REMARKS

In view of the following remarks, Applicant respectfully requests that the application be forwarded on to issuance.

Rejections under § 103

Claims 1-7, 10-11, 13-16, 18, 20-26, 29-30, 32-35, and 37 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Publication No. 2003/0236775 to Patterson et al. ("Patterson") in view of U.S. Patent 5,896,495 to Stein et al. ("Stein").

Response to § 103 Rejections

In making out the rejection of claim 23, which recites: "a method according to claim 20, wherein one of the selectable search strategies is a flag-based attribute search strategy that involves comparing a specified attribute value and qualifying flag information with information associated with the user interface object", the Office states:

> **Final Office Action, Page 4, Paragraph # 9** (Emphasis Added)
> As to claim 23, ***Patterson as modified does not specifically teach wherein one of the selectable search strategies is a flag-based attribute search strategy.*** However, Patterson disclosed a unique name string that contains a predetermined set of properties of a GUI object, the properties of the GUI object in the unique name are organized such that it is easy to extract the property names from the string name of the GUI object, and determine the closest matching GUI objects based on the properties matched [pgs. 4-5, paragraphs 49-63]. ***It would have been obvious to one of ordinary skill in the art at the time the invention was made, to have recognized that the teaching of Patterson involves the comparison of various attribute value and flag information***.

Applicant agrees that Patterson **does not teach wherein one of the selectable search strategies is a flag-based attribute search strategy**. But Applicant respectfully disagrees with the Office's assertion that it would be obvious "to have recognized that the teaching of Patterson involves the comparison of various attribute and flag information". In the first place, it is important to note that the word "flag" and the term "flag-based search" do not occur anywhere in Patterson. Therefore, at a minimum the Office has not met its burden of showing that Patterson contemplates the use of flags to do a search because the Office has made no showing from Patterson and provided no other evidence.

Also, and in contrast to the Office's position, Patterson creates "unique string reference, or name, for the object." (Patterson, paragraph 51.) Then, "[w]hen the test script is executed on the GUI application, both of which are running at the same time, the 'name' of the object the user wishes to manipulate, i.e. the object identified in the test script, is converted into a format comparable to the runtime GUI object database." (Patterson , paragraph 61.) Finally, the object name is "***looked up*** in the runtime GUI object database." (Id, Emphasis Added.) To summarize, Patterson creates a "unique string" for an object, and then uses the string to search for the object in the GUI object database.

Claim 23, on the other hand, recites a "flag-based attribute search." Applicant's specification, for example, teaches that the flags "qualify the search for the attributes in different ways depending on the nature of the flags. That is, the flags can be viewed as ***performing a filtering function*** in the search … ***by restricting the search results to a certain subset of results depending on the nature of the flags***." (Applicant's Specification, page 27, paragraph 93, Emphasis added.) As an example of various flags that can be used in a search, an excerpt from Applicant's specification is reproduced below:

Applicant's Specification, Paragraph 94, Table 5

[0094] **Table 5: Search Flags**

Flag(s)	Corresponding Object Attribute
XA_USE_TEXT XA_USE_ANCESTORTEXT	These flags instruct the machine to perform the search within the text of a window-type object under consideration, or within the text in an ancestor object in the hierarchy of objects.
XA_USE_CLASS XA_USE_ANCESTORCLASS	These flags instruct the machine to perform the search within ClassName related information.
XA_DIALOGS	N/A
XA_USE_MODULE	This flag instructs the machine to perform the search within ModuleName related information.

Flag(s)	Corresponding Object Attribute
XA_USE_PROCESSID	This flag instructs the machine to perform the search in ProcessID related information.

As shown above, all of the search flags provide instructions to the machine that tell the machine how to perform the search. The search flags reproduced above are just one example of flags. Other flags, for example, include the flag "T_F_INCLUDE_EDIT", which instructs "the machine to include Edit controls text in the search." (<u>Applicant's Specification</u>, Page 37, Table 7.) Similarly, the flag "T_F_INCLUDE_BUTTONS" instructs "the machine to include text from all types of buttons". (<u>Applicant's Specification</u>, Page 39, Table 8.) Thus, these flags perform a filtering function in the search.

In order to further assist the Office in understanding one of many benefits of a "flag-based" search, an example from Applicant's Specification is reproduced below:

> **<u>Applicant's Specification, Page 29, Paragraphs 96-98</u>**
>
> [0096] *Example (a).*
>
> [0097] The following example presents a technique for finding the word "CompanyX" in a UI object using the Find method. The Find method in this example is accessed through the Manager object (as indicated by the notation "oManger.Find()."
>
> //JSCRIPT sample using Find and combining flags with attributes as part of XAble.Manager
>
> var UIFound = oManager.Find (XA_USE_TEXT, "*CompanyX*");
>
> [0098] In this case, the flag XA_USE_TEXT instructs the XABLE tool that it is to locate the word "CompanyX" in the text portion of UI objects provided by the test application.

As shown above, "the flag XA_USE_TEXT instructs the XABLE tool that it is to ***locate the word 'CompanyX' in the <u>text portion</u> of UI objects*** provided by the test application." (<u>Patterson</u>, Page 29, Paragraph 98, Emphasis Added.) In other words, the flag XA_USE_TEXT instructs the machine to perform a word search in a specific location, which in this case is the "text portion".

Patterson does not teach or in any way suggest that one of the selectable search strategies is a ***flag-based attribute search strategy*** that involves comparing a specified attribute value (in the example above: the word "CompanyX") and ***qualifying flag information*** (in the example above: XA_USE_TEXT, which instructs the machine to search in the "text portion") with information associated with the user interface object. As noted above, Patterson never mentions the term "flag", and instead teaches a search based on a unique string name. Furthermore, Patterson does not contemplate a flag that performs a filtering function in the search. Because Patterson deals solely with "string-based" searches, and never mentions flags, it would not be obvious to modify Patterson to include flags. To this extent, Stein adds nothing of significance.

Applicant maintains all of the arguments advanced in the previous Office Action response, however, in order to advance prosecution, all of the independent claims have been amended to clarify that "one of the selectable search strategies is a flag-based attribute search strategy that involves comparing a specified attribute value and qualifying flag information with information associated with the user interface object, wherein the flag information is configured to provide a filtering function in a search". Claim 1 has been amended to include the additional limitation: "…by restricting results of the search to a certain subset of results based on the flag information". Applicant therefore respectfully requests that the Office withdraw its rejections of independent claims 1, 10, 18, 20, 29, and 37.

Applicant also submits that, should the Office intend to maintain its rejection on this basis, that the Office must provide a reference and reasoning in addition to its assertion that it would have been obvious.

Applicant respectfully submits that the dependent claims relying on these independent claims are allowable as relying on an allowable base claim and for their own recited features.

Conclusion

All of the claims are in condition for allowance. Accordingly, Applicant requests that the Office issue a Notice of Allowability. If the Office's next anticipated action is to be anything other than issuance of a Notice of Allowability, Applicant respectfully requests a telephone call for the purpose of scheduling an interview.

Respectfully Submitted,

Dated: 5 December 2007

By: /Michael K. Colby/
Michael K. Colby
Reg. No. 45,816
(509) 755-7254